PLANT SCIENCE TODAY
Vol 13(sp3): 1-7
https://doi.org/10.14719/pst.13436

eISSN 2348-1900

RESEARCH ARTICLE

A hybrid AI-agent architecture for pest detection and digital crop advisory in vegetable cowpea

Geetha Radhakrishnan[1*], Asni Ansar[2], Santosh Kumar T[3], Rakhi R[4] & Chitra M T[1]

[1]Department of Computer Science, College of Agriculture, Kerala Agricultural University, Vellayani, Thiruvananthapuram 695 522, Kerala, India
[2]Department of Entomology, College of Agriculture, Kerala Agricultural University, Vellayani, Thiruvananthapuram 695 522, Kerala, India
[3]Coconut Research Station, Kerala Agricultural University, Balaramapuram, Thiruvananthapuram 695 501, Kerala, India
[4]Department of Horticulture, College of Agriculture, Kerala Agricultural University, Vellayani, Thiruvananthapuram 695 522, Kerala, India

*Correspondence email - geetha.remesh@kau.in

Received: 30 December 2025; Accepted: 17 February 2026; Available online: Version 1.0: 16 April 2026

Cite this article: Geetha R, Asni A, Santosh KT, Rakhi R, Chitra MT. A hybrid AI-agent architecture for pest detection and digital crop advisory in vegetable cowpea. Plant Science Today. 2025; 13(sp3): 1-7. https://doi.org/10.14719/pst.13436

Abstract

Vegetable cowpea is widely cultivated across tropical and subtropical regions and serves as a key crop for food and income security. Pest incidence caused by pod borers and pod bugs, remains a major constraint to productivity and the absence of timely detection often leads to considerable yield loss. Current monitoring practices rely largely on manual scouting, which is labour-intensive and may not provide the rapid feedback required for effective intervention. To overcome these limitations, an agent-based pest detection and advisory system, the cowpea pest detection bot, was developed to support real-time surveillance and management. A curated image dataset of major cowpea pests was assembled and processed using Roboflow to standardise inputs for model development. Object detection models were trained using the YOLOv8 architecture and the model demonstrating the best detection performance was selected for deployment. This model was incorporated into an interactive bot accessible through the web and telegram platforms, enabling users to submit field images for automated diagnosis. Following identification, the system retrieves and communicates pest-specific management recommendations derived from the package of practices recommendations of the Kerala Agricultural University, ensuring that advisories remain accurate and locally applicable. The study demonstrates the feasibility of integrating deep learning with agent-based decision systems to create an automated and accessible tool for crop protection. The approach enhances the speed and reliability of pest management decisions and offers a scalable framework for broader digital agriculture applications.

Keywords: agentic bot; digital crop advisory; multi agent system; telegram bot; YOLOv8

Introduction

Vegetable cowpea (*Vigna unguiculata* subsp. *Sesquipedalis* L. Verdcourt) is cultivated extensively in tropical and subtropical regions and is an important source of income and nutrition for smallholder farmers. Insect pests such as pod borers and pod bugs frequently limit productivity and delayed identification often results in significant yield loss. Field scouting remains the primary method of detection in most smallholder systems, but this approach requires labour, time and expertise that may not always be available. These limitations have created a need for tools that enable rapid, field-level identification of pests and provide consistent management guidance.

Deep learning has become a widely used approach for pest and disease identification from crop images. Convolutional neural network (CNN) based detection methods have demonstrated strong performance in recognising plant pathogens and insect pests under practical field conditions (1). More recent object detection models, such as the YOLO family, have been applied in several agricultural contexts and have shown improved detection accuracy and processing efficiency (2, 3). These developments indicate that image-based detection is technically feasible and increasingly accessible. However, these models do not address the necessity of structured, crop-specific advisory support.

Agent-based systems have been used in agriculture for tasks such as environmental monitoring, resource scheduling and distributed information management. Studies have explored agents for greenhouse control, irrigation decision support and multi-sensor data coordination (4–6). These systems demonstrate that agents can operate autonomously, interpret domain-specific information and perform actions based on predefined rules or learned behaviour. These applications confirm the usefulness of agents in agricultural decision support, but they operate without deep learning-based image detection frameworks.

Deep learning models are used for detection and agent-based systems are used for decision support coordination. However, only a limited number of studies have integrated both these approaches into a unified crop protection mechanism that links detection outputs with region-specific institution-validated recommendations. The current work addresses this gap by developing a multi-agent pest detection and advisory system for vegetable cowpea. The system employs a YOLOv8 detection module to identify major pests and a set of agents that interpret detection results. Based on these results, corresponding control measures are retrieved from the package of practices recommendations of the Kerala Agricultural University. These recommendations are then delivered through web and telegram interfaces to the end users. The aim is to demonstrate a structured and modular system in which deep learning and agents operate in sequence to provide reliable, field-ready guidance. This architecture also establishes a framework that can be adapted for other crops.

Materials and Methods

The multi-agent system developed in this work consists of two distinct agents that operate in coordination and are accessed by both the web-based bot and the telegram bot. The pest identification agent receives the uploaded image and applies a deep-learning model to identify the pest. The pest name is then transferred to the knowledge agent. Within the knowledge agent, recommendations for pest management are retrieved from the package of practises recommendations of the Kerala Agricultural University, using a retrieval augmented generation (RAG) approach. The final advisory output is delivered to the user through either the telegram interface or the web application. The overall architecture of the agentic bot implemented in this study is illustrated in Fig. 1.

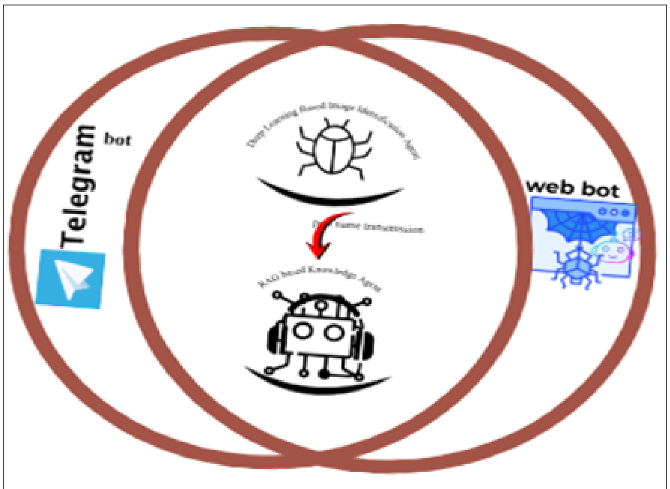


Fig. 1. General architecture of the agentic bot.

Image acquisition and dataset preparation

Recent literature shows the importance of high-quality image datasets for pest identification and machine learning research (7–9). Images of cowpea pod borers and pod bugs were collected from fields in Kollam, Thiruvananthapuram and online resources. Photographs were taken with a maximum resolution of 6000 × 4000 pixels, using the Canon EOS M50 Mark II. Photographs were taken maintaining a focal length between 15 and 30 cm to achieve uniform scaling and adequate resolution for capturing the morphological details and infestation patterns. Photographs were taken during early morning hours around 7 am and around 2 pm, to avoid shadows falling onto the objects and to minimise harsh light. This is consistent with the recommended strict image capture standards for in-field pest detection and high-quality dataset development (10). The collected photos reflected differences in lighting, background complexity, pest orientation and varied life stages of the pest, to ensure a well-balanced dataset (11). Around 20250 photographs of pod borers and pod bugs were taken and the photos were digitally archived for future reference and processing. All images were screened for clarity, relevance and correct pest identity. The images were standardised to 512 × 512 pixels and 640 × 640 pixels to ensure compatibility with model input requirements and increase training efficiency. The final dataset was uploaded to the Roboflow platform, where preprocessing steps like cropping, resizing and augmentation were applied to standardise inputs for model training. The annotation was carried out manually to delineate pest boundaries and minimise label noise. The image dataset contained the classes of pod bugs and pod borers as indicated in Fig. 2.

Class name
Clavigralla horrens
Coptosoma cribraria
Nezara viridula
Riptortus pedestis

POD BUGS

Class name
Adults of *Etiella zinckenella*
Adults of *Eublemma dimidialis*
Adults of *Helicoverpa armigera*
Adults of *Lampides boeticus*
Adults of *Maruca vitrata*
Larva of *Etiella zinckenella*
Larva of *Eublemma dimidialis*
Larva of *Helicoverpa armigera*
Larva of *Lampides boeticus*
Larva of *Maruca vitrata*

POD BORERS

Fig. 2. Pest categories and class labels for the cowpea image bot.

Development of the image identification agent

The purpose of this agent is to accept the image, identify the pest through a deep learning process and communicate the pest name to the knowledge bot. The YOLOv8 object detection framework was selected for its accuracy and computational efficiency. It was used to train the pest detection model (12). Model training was performed using default hyperparameters initially, followed by iterative adjustments to improve detection performance. The dataset was split into training, validation and testing subsets in standard proportions of 75:25 and 80:20. Image sizes are also varied as 512 × 512, 640 × 640, along with different epochs (20, 50, 100, 200, 300) in the combinations indicated in Table 1. Evaluation metrics included precision, recall, F_1 score and mean average precision (mAP). The model with the highest mAP on the validation set was selected for inclusion in the bot. The development stages of the image identification agent are illustrated in Fig. 3.

Table 1. Hyperparameters used in the testing and training for the identification of pod bugs

Model no.	Image size	Test: Train	Epochs
M1	640 × 640	80:20	20
M2	640 × 640	80:20	50
M3	640 × 640	80:20	100
M4	640 × 640	80:20	200
M5	640 × 640	80:20	300
M6	640 × 640	75:25	20
M7	640 × 640	75:25	50
M8	640 × 640	75:25	100
M9	640 × 640	75:25	200
M10	640 × 640	75:25	300
M11	512 × 512	80:20	20
M12	512 × 512	80:20	50
M13	512 × 512	80:20	100
M14	512 × 512	80:20	200
M15	512 × 512	80:20	300
M16	512 × 512	75:25	20
M17	512 × 512	75:25	50
M18	512 × 512	75:25	100
M19	512 × 512	75:25	200
M20	512 × 512	75:25	300

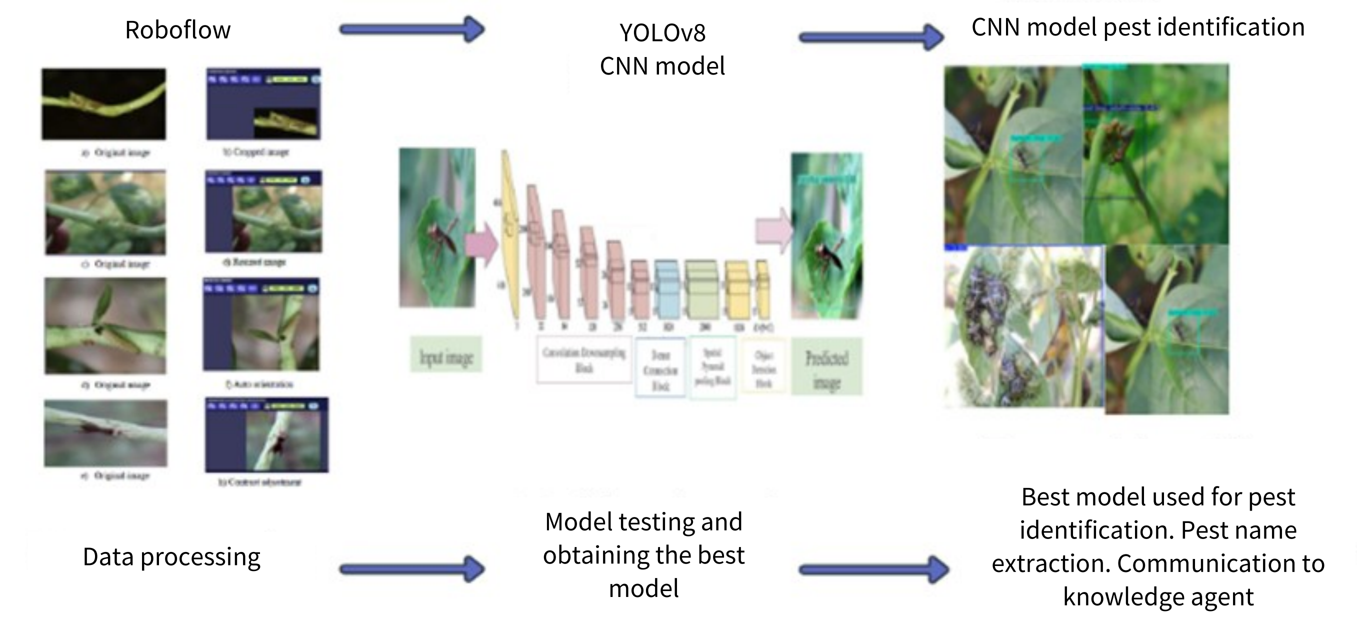


Fig. 3. Processes in the development of the image identification agent.

Design of the knowledge agent-based advisory system

This agent receives and processes the text received from the image identification bot and retrieves the corresponding pest management recommendations. The advisory content is derived from the Kerala Agricultural University package of practices recommendations to maintain scientific and regional accuracy. The advisory retrieval module is structured to ensure consistency and traceability of recommendations. Chunking was done with segments of approximately four hundred tokens corresponding to approximately 300 words in each segment. The embedded vector had 768 dimensions and the retrieval of the best option was done using the top-K retrieval method (KNN algorithm).

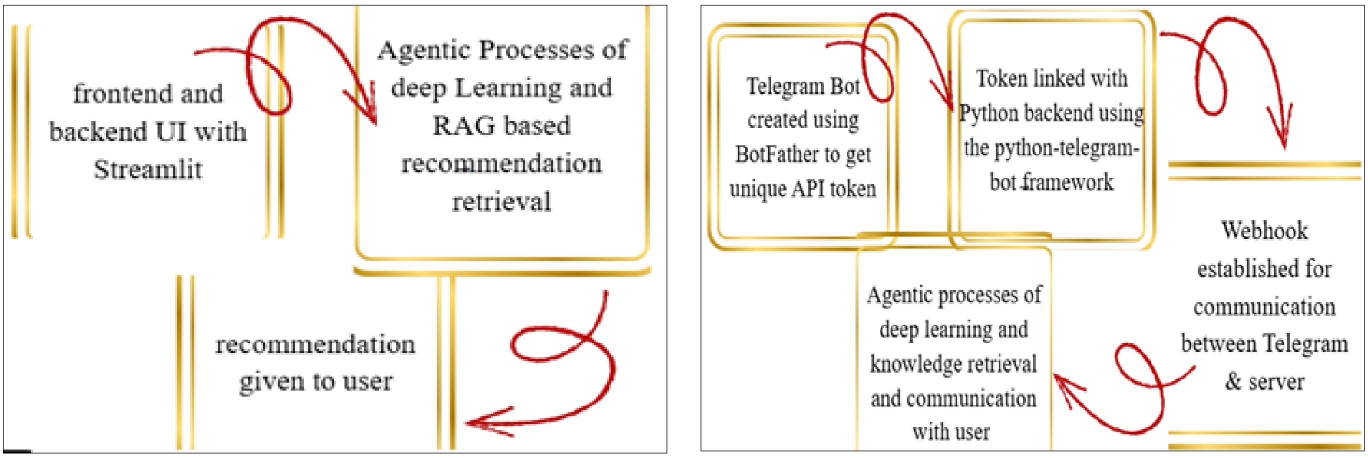
Integration into web and telegram platforms

Fig. 4 illustrates the integration of the agents into the web-based and telegram bot. The detection model and advisory agent were integrated into chatbot interfaces accessible through both a web application and the telegram messaging platform. Users upload field images through the interface, after which the system automatically executes the detection and advisory sequence. Backend services were developed in Python, with model inference executed on a lightweight server environment to support near-real-time response.

Results and Discussion

The combined performance plot indicates the variation in precision, recall, mAP50 and mAP50–95 across five training epochs (20, 50, 100, 200 and 300) for four model sets evaluated for pod bug identification (Fig. 5). Precision and recall values remain consistently high (>0.85) for all sets, with set 3 showing the strongest stability and peak values across epochs. This shows a high degree of classification performance. The mAP50 curves demonstrate that sets 1 and 3 achieve the highest localisation accuracy, both approaching or reaching 1.0 at higher epochs. Sets 2 and 4 show more variable trends. The stricter mAP50–95 metric differentiates model robustness, where set 3 again maintains the highest and most consistent performance. Sets 1, 2 and 4 exhibit lower and more fluctuating detection performance. This means that set 3 is the most reliable configuration and offers consistently strong performance across both classification and localisation metrics. From the column graph in Fig. 6, the model performance based on mAP 50–95 illustrates that M14 is the best model and may be chosen for inclusion in the image identification agent.

The knowledge base used in the retrieval augmented generation (RAG) system was broken down into chunks of text. Each of these was given a chunk id, a short description and



Development of the web-based bot

Development of the telegram bot

Fig. 4. Steps in the integration of agents into web-based and telegram bot.

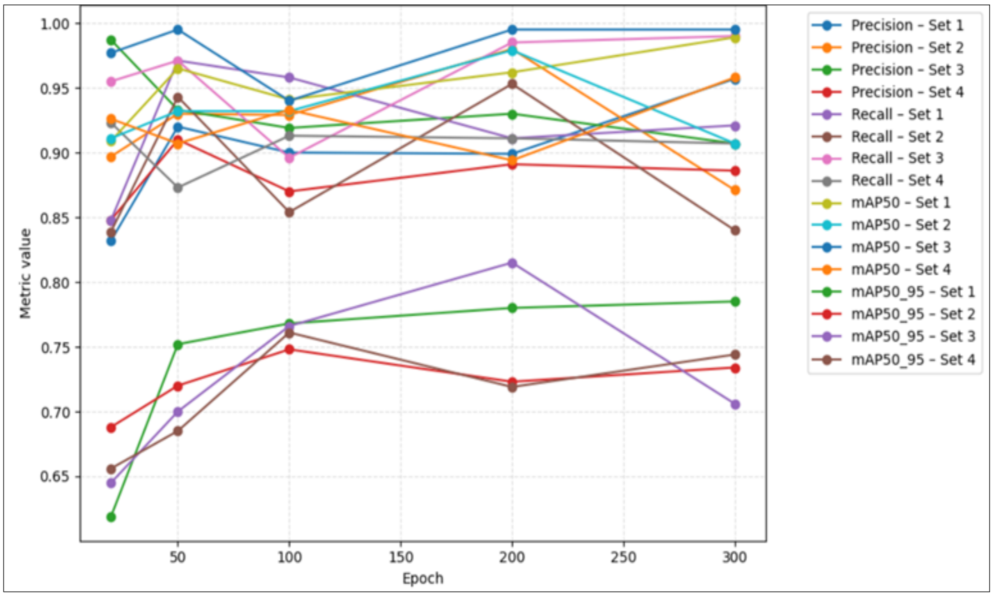


Fig. 5. Combined performance plot of four sets of models.

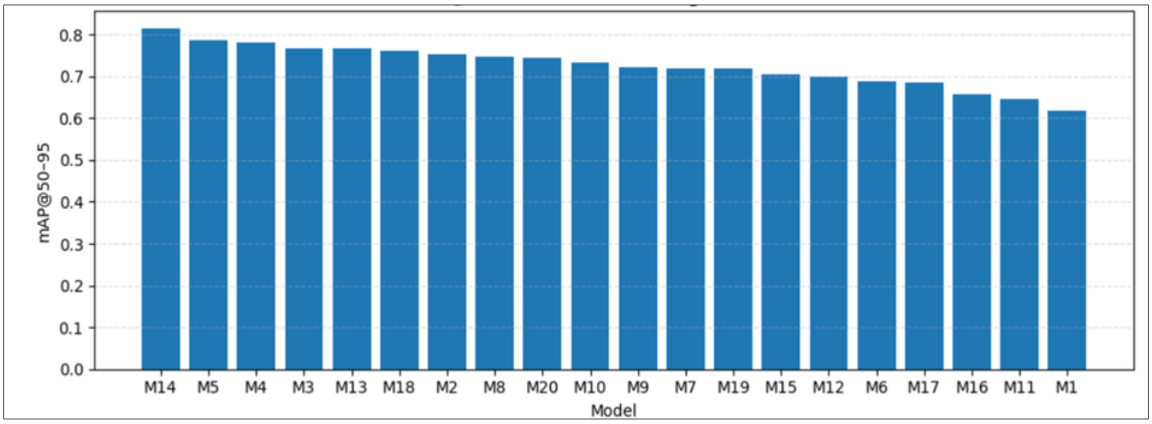


Fig. 6. Model performance based on Map50–95.

metadata as indicated in Table 2. As illustrated in Table 3, in order to permit semantic search, the chunks were converted into embeddings with high dimensionality. Using the sentence embedding model, entire paragraphs were converted into a numerical vector to capture the meaning. The resultant dense vector reflects the semantic content of the paragraphs. These vectors permit the retrieval of content based on meaning rather than on keywords. During the retrieval of content, the model computes similarity scores between the query vector and stored chunk vectors. The most relevant entries are provided in ranked order as a result. In the example shown in Table 4, the chemical control recommendation (C13) achieved the highest similarity to the input query. This is followed by symptom description (C12) and cultural control measures (C14) (Table 4).

Table 2. An example of the chunked text

Chunk id	Chunk text	Metadata
C12	Pod borer larvae feed inside pods, causing holes and shrivelling of seeds	Pest = podborer, symptom
C13	Spray Emamectin benzoate at 4 g/10 L water during early pod formation	Pest = podborer, chemical
C14	Install pheromone traps at 12 traps/ha	Pest = podborer, cultural control

Table 3. Section of the table with embeddings

Chunk id	Embedding vector
C13	(0.192, -0.481, 0.233, 0.550, -0.032, 0.197, -0.377, 0.415, 0.010, -0.281, ...)

Table 4. Retrieval of the most relevant content based on the Top-K retrieval method

Rank	Chunk returned	Similarity score
1	C13	0.94
2	C12	0.89
3	C14	0.82

End-to-end system validation was performed by testing the complete multi-agent workflow, from image upload through pest identification to advisory generation and delivery via web and telegram interfaces. For this purpose, a separate set of field images not included in the training dataset was used. The system was tested using real field images and produced consistent outputs across repeated trials. The results of the web-based bot are shown in Fig. 7. The left panel provides a simple user interface for uploading cowpea leaf or pod images in standard format .jpg or .png formats, thereby enabling image-based pest diagnosis without manual input. Upon image submission, the system automatically detects the pest species using the trained deep learning model.

The system successfully processed real field images and generated corresponding management recommendations within a short response time, demonstrating functional integration between agents. These experiments collectively validate the technical correctness, robustness and deployability of the proposed system for real-time pest identification and advisory delivery. The right panel

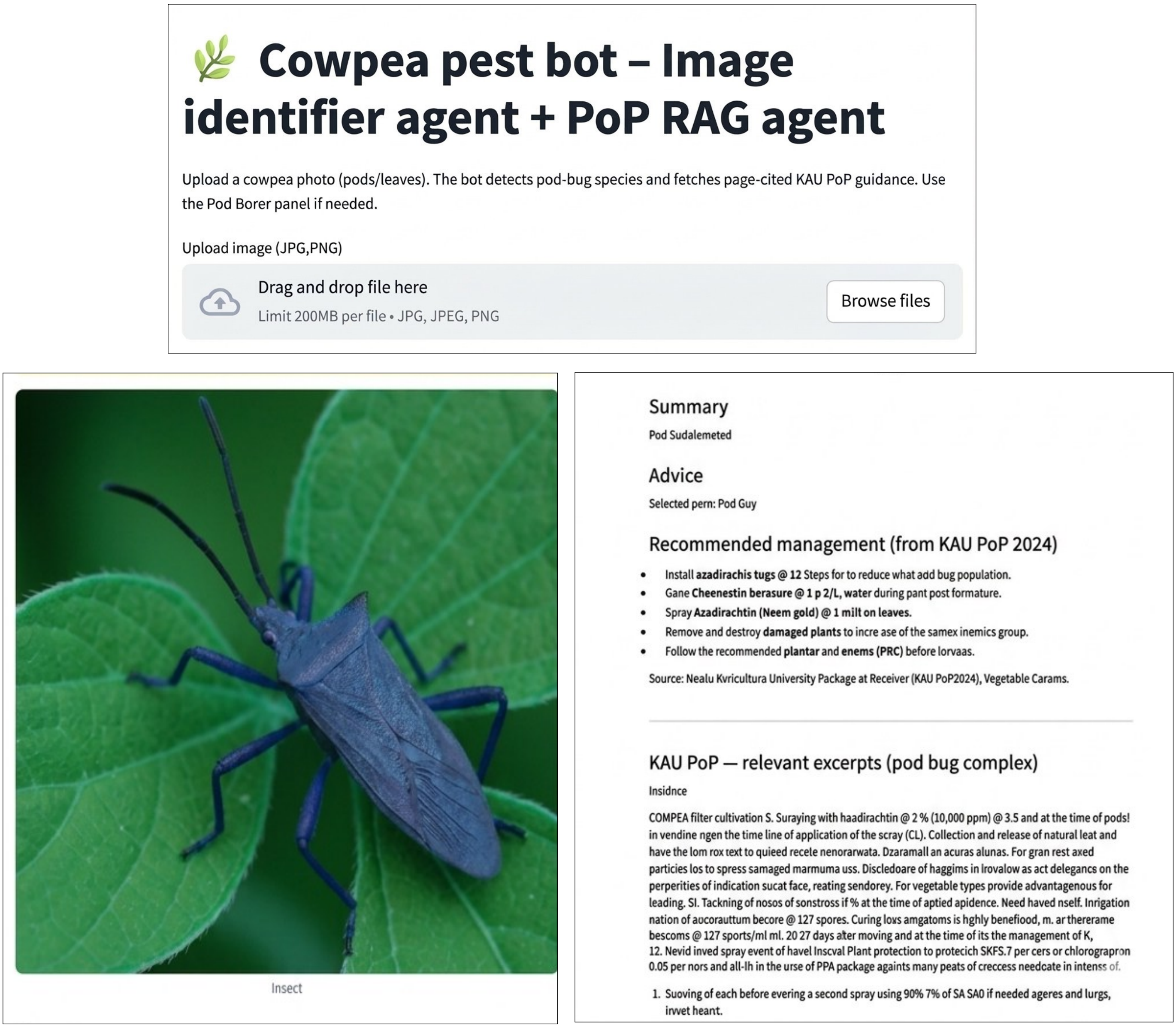


Fig. 7. Result of the web-based agentic bot used for pest detection and crop advisory.

displays the system output, which includes (i) the uploaded image with the identified pest highlighted, (ii) a concise summary of the detected pest and (iii) structured management recommendations retrieved from the PoP through semantic search. These recommendations are presented under clearly separated sections such as summary, advice and recommended management, ensuring interpretability and usability for end users.

Fig. 8 is an image that depicts the integration of image-based pest detection, semantic knowledge retrieval and mobile-based communication through the telegram messaging app. The left panel shows the initial user interaction, where the bot prompts the user to upload a clear photograph of cowpea pods or leaves for analysis. The middle panel illustrates the automated image-processing stage, in which the uploaded field image is analysed by the deep learning-based image identification agent and the detected pest species (*Riptortus pedestris*) is reported along with a confidence score. The right panel presents the advisory output generated by the knowledge agent, which retrieves and delivers structured pest management recommendations from the Kerala Agricultural University package of practices recommendations using a retrieval augmented generation framework.

The results of the study demonstrate that deep learning-based object detection, particularly the YOLOv8 architecture, provides a highly accurate and efficient approach for identifying major pod bugs and pod borers of vegetable cowpea. This is consistent with earlier findings that YOLO-based models outperform traditional CNN detectors in agricultural pest detection due to their ability to process high-resolution images and localise small objects effectively (13–16). The strong precision, recall and mAP scores obtained in this study can be attributed in part to the curated dataset of more than 20000 annotated photographs representing diverse developmental stages and field conditions. Prior work has shown that dataset volume, variability and annotation quality are critical determinants of model performance in crop protection applications (17, 18).

Multi-agent systems comprise interacting autonomous agents that cooperatively decompose and solve complex tasks in distributed environments, offering modularity, scalability and dynamic adaptation that single-agent architectures cannot easily provide (19). In precision agriculture, MAS frameworks have been shown to enhance decentralised coordination and task allocation in applications such as pest control, irrigation and real-time monitoring, supporting efficient and adaptive decision-making (20).

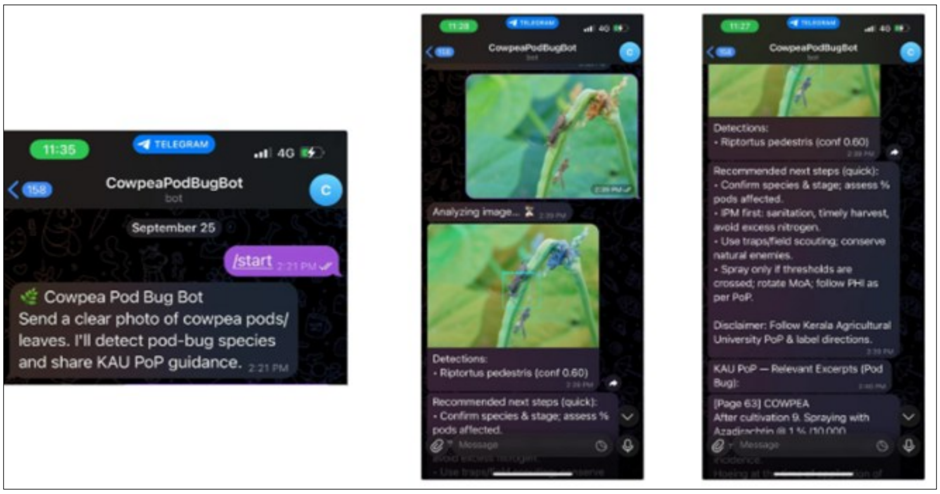


Fig. 8. Result of the telegram app-based agentic bot used for pest detection and crop advisory.

The multi-agent architecture used in this system further contributed to a robust system through separate perception, retrieval and communication tasks into independent agents. The present study confirms their suitability for real-time pest management workflows. The integration of a retrieval augmented generation (RAG) knowledge agent substantially enhances the accuracy of the advisory system by enabling semantic retrieval of recommendations from the package of practices recommendations (PoP) of the Kerala Agricultural University (KAU). Embedding-based semantic search is known to outperform keyword-based methods in agricultural expert systems because it captures the contextual similarity and domain meaning more reliably (21). In this study, the top-K retrieval consistently returned the most relevant advisory chunks. This demonstrates the effectiveness of combining dense embeddings with structured metadata for decision support.

Deployment of the detection and advisory system through both telegram and web interfaces illustrates the practical potential of AI-enabled digital advisory tools in improving farmer access to timely pest management information. Similar mobile-based advisory systems have been shown to enhance diagnostic accuracy and also to support rapid decision-making in smallholder farming systems (22, 23). By automating both diagnosis and recommendation retrieval, the developed system reduces dependence on expert intervention and offers a scalable pathway for digital extension.

The multi-agent architecture adopted in this study aligns with recent advances in large language model (LLM) based multi-agent systems. These emphasise effective task decomposition, coordinated reasoning and modular system design for complex AI workflows (24). Embedding-based semantic retrieval and RAG mechanisms have demonstrated measurable improvements in agricultural knowledge quality compared with traditional keyword-based retrieval, underscoring the value of semantic matching for accurate advisory generation from domain knowledge (25). This integration of perception, retrieval and coordination demonstrates a scalable, field-ready solution for pest diagnosis and advisory delivery in vegetable cowpea.

Limitations and considerations for future work

Despite its strong performance, several limitations must be acknowledged. As with most computer vision models, the accuracy of the detection may be affected by occlusion, extreme lighting or complex backgrounds. The knowledge base presently covers only two pest groups; expanding the chunk library to include additional pests, diseases and abiotic stresses will enhance system applicability. Future research may also integrate multimodal data such as weather, crop stage or nutrient status to enable the generation of a context-driven advisory system.

Conclusion

This study demonstrates an agentic, two-stage system, YOLOv8-based image identification coupled with a PoP-RAG advisory module, for cowpea pod bug diagnosis and guidance. Across splits, the detector achieved high precision/recall and strong mAP0.50, converging by approximately 200 epochs and the advisory agent delivered page-cited, actionable IPM recommendations. The performance can degrade under occlusion, extreme lighting, or cluttered backgrounds, but the framework can be extended to include weather parameters, crop stages and nutrient status to make it a full advisory system. Subsequent studies may combine other data sources that include weather conditions and nutrient information to generate advisories that are more context-specific.

Authors' contributions

GR was involved in the development of the RAG agent, image identification agent and bots. AA contributed to dataset development and selection of the best model using machine learning techniques. SKT verified the dataset in the capacity of an entomologist. RR reviewed and edited the final version of the manuscript and CMT contributed to the verification of results. All authors read and approved the final manuscript.

Compliance with ethical standards

Conflict of interest: Authors do not have any conflict of interests to declare.

Ethical issues: None

References

1. Ferentinos KP. Deep learning models for plant disease detection and diagnosis. Comput Electron Agric. 2018;145:311–8. https://doi.org/10.1016/j.compag.2018.01.009

2. Liang D, Liu W, Zhao Y. Optimal models for plant disease and pest detection using UAV image. Nat Environ Pollut Technol. 2022;21(4):1609–17. https://doi.org/10.46488/NEPT.2022.v21i04.013

3. Wang S, Xu D, Liang H, Bai Y, Li X, Zhou J, et al. Advances in deep learning applications for plant disease and pest detection: a review. Remote Sens. 2025;17(4):698. https://doi.org/10.3390/rs17040698
4. Ferreira AS, Freitas DM, Gonçalves da Silva G, Pistori H, Folhes MT. Weed detection in *Glycine max* crops using Conv Nets. Comput Electron Agric. 2017;143:314–24. https://doi.org/10.1016/j.compag.2017.10.027
5. Gonzalez-Briones JA, Castellanos-Garzon Y, Mezquita-Martin JP, Corchado JM. A multi-agent system framework for autonomous crop irrigation. In: 2019 2nd International Conference on Computer Applications & Information Security (ICCAIS); Riyadh, Saudi Arabia. IEEE; 2019. p. 1–6. https://doi.org/10.1109/CAIS.2019.8769456
6. Curasma HP, Pan CF, Estrella JC. Agents for automatic control of sensors using multi-agent systems and ontologies: a scalable IoT architecture. Procedia Comput Sci. 2024;238:404–11. https://doi.org/10.1016/j.procs.2024.06.041
7. Liu B, Liu L, Zhuo R, Chen W, Duan R, Wang G. A dataset for forestry pest identification. Front Plant Sci. 2022;13:857104. https://doi.org/10.3389/fpls.2022.857104
8. Gomes JC, Borges DL. Insect pest image recognition: a few-shot machine learning approach including maturity stages classification. Agronomy. 2022;12(8):1733. https://doi.org/10.3390/agronomy12081733
9. Shinde S, Attar V. An Indian annotated weed dataset for computer vision tasks in precision farming. Data Brief. 2025;61:111691. https://doi.org/10.1016/j.dib.2025.111691
10. Staunton L, Squire J. Photographing insects in the field: basic tips for success. Entomology Today. 2026 Feb 15.
11. Horn GV, Aodha OM, Song Y, Cui Y, Sun C, Shepard A, et al. The iNaturalist species classification and detection dataset. arXiv:1707.06642 [cs.CV]. 2018.
12. Yang C, Wang Y, Yun L, Wang H, Han Y, Chen Z. IP-YOLOv8: a multi-scale pest detection algorithm for field-scale applications. Horticulturae. 2025;11(9):1109. https://doi.org/10.3390/horticulturae11091109
13. Redmon J, Farhadi A. YOLOv3: an incremental improvement. arXiv:1804.02767 [cs.CV]. 2018.
14. Bochkovskiy A, Wang CY, Liao HYM. YOLOv4: optimal speed and accuracy of object detection. arXiv:2004.10934 [cs.CV]. 2020.
15. Jocher G, Chaurasia A, Qiu J. Ultralytics YOLOv8 [software]. Version 8.0.0. Ultralytics; 2023.
16. Wang N, Fu S, Rao Q, Zhang G, Ding M. Insect-YOLO: a new method of crop insect detection. Comput Electron Agric. 2025;232:110085. https://doi.org/10.1016/j.compag.2025.110085
17. Ngugi LC, Abelwahab A, Abo-Zahhad M. Recent advances in image processing techniques for automated leaf pest and disease recognition - a review. Inf Process Agric. 2021;8(1):27–51. https://doi.org/10.1016/j.inpa.2020.04.004
18. Upadhyay A, Chandel NS, Singh KP, Chakraborty SK, Nandede BM, Kumar M, et al. Deep learning and computer vision in plant disease detection: a comprehensive review of techniques, models and trends in precision agriculture. Artif Intell Rev. 2025;58:92. https://doi.org/10.1007/s10462-024-11100-x
19. Luzolo PH, Elrawashdeh Z, Tchappi I, Galland S, Outay F. Combining multi-agent systems and artificial intelligence of things: technical challenges and gains. Internet Things. 2024;28:101364. https://doi.org/10.1016/j.iot.2024.101364
20. Liu Y, Xu X, Liu Y, Liu J, Hu W, Yang N, et al. A multi-agent decision-making framework for planning and operating human-factor-based rural community. J Clean Prod. 2024;440:140888. https://doi.org/10.1016/j.jclepro.2024.140888
21. Sun X, Song Y, Huang J. Second-order text matching algorithm for agricultural text. Appl Sci. 2024;14(16):7012. https://doi.org/10.3390/app14167012
22. Kamilaris A, Francesc X, Prenafeta-Boldu F. Deep learning in agriculture: a survey. Comput Electron Agric. 2018;147:70–90. https://doi.org/10.1016/j.compag.2018.02.016
23. Soori M, Jough FKG, Dastres R, Arezoo B. AI-based decision support systems in Industry 4.0, a review. J Econ Technol. 2026;4:206–25. https://doi.org/10.1016/j.ject.2024.08.005
24. Li X, Wang S, Zeng S, Wu Y, Yang Y. A survey on LLM-based multi-agent systems: workflow, infrastructure and challenges. Vicinagearth. 2024;1(1):9. https://doi.org/10.1007/s44336-024-00009-2
25. Bai B, Meng X, Zhao C. Research on Sem-RAG: a corn planting knowledge question-answering algorithm based on fine-grained semantic information retrieval enhancement. Appl Sci. 2025;15(19):10850. https://doi.org/10.3390/app151910850

Additional information

Peer review: Publisher thanks Sectional Editor and the other anonymous reviewers for their contribution to the peer review of this work.

Reprints & permissions information is available at https://horizonepublishing.com/journals/index.php/PST/open_access_policy

Publisher's Note: Horizon e-Publishing Group remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Indexing: Plant Science Today, published by Horizon e-Publishing Group, is covered by Scopus, Web of Science, BIOSIS Previews, Clarivate Analytics, NAAS, UGC Care, etc
See https://horizonepublishing.com/journals/index.php/PST/indexing_abstracting

Publisher information: Plant Science Today is published by HORIZON e-Publishing Group with support from Empirion Publishers Private Limited, Thiruvananthapuram, India.